

10/673,805

MS301930.01/MSFTP381US

**AMENDMENTS TO THE SPECIFICATION****In the Specification:**

Please replace the paragraph beginning on page 1, line 1 that starts with "Title: CONCURRENT" with the following amended paragraph:

Title: CONCURRENT ARBITRATION OF MULTIDIMENSIONAL REQUESTS FOR INTERRUPT RESOURCES

Please replace the paragraph beginning on page 13, line 28 that starts with "Figs. 5 and 6 illustrate" with the following amended paragraph:

Figs. 5 and 6 illustrate an exemplary interrupt arbitration system in accordance with an aspect of the present invention. A multiprocessor architecture is depicted having a processor at 510 and 520, wherein the respective processors are shown with associated IDT range entries and possible IDT range entries which are not presented to an arbiter system library (not shown). In addition the processor may interact with interrupt controllers at 530 that receive interrupt inputs from optional link nodes at 540, wherein the link nodes operate as interrupt multiplexers that connect multiple interrupt sources to a single interrupt on the interrupt controllers 530. The link nodes 540 receive input from one or more IRQ user data sources at 550 of Fig. 5 and at 560 From Fig. 6. These sources contain such information as processor information, interrupt controller information, link node connection information, and MSI (Message-Signaled Interrupt) target information. As illustrated the sources at 550 are associated with allocation ranges depicted at 570 of Fig. 6, and the sources at 560 [[570]] of Fig. 6, are associated with possible allocation ranges at 580 of Fig. 6.

10/673,805

MS301930.01/MSFTP381US

Please replace the paragraph beginning on page 14, line 13 that starts with "Fig. 7 illustrates" with the following amended paragraph:

Fig. 7 illustrates an example operating system environment 700 in accordance with an aspect of the present invention. In the system 700, an application 710 having a configuration manager 714 for device-specific data, and a Windows Management Infrastructure (WMI) component 718 for machine-specific data interact via remote procedure calls (RPC) with a COM server at 730 and a PnP ~~manager~~ manger service at 734. These components 730 and 734 interact across a user mode/kernel mode boundary 740 with a kernel-mode WMI manager 750 and a kernel-mode PnP manager 754. The manager's 750 and 754 interact with an IRQ arbiter 760 that is associated with a driver component 764. As described above, interrupt resources are negotiated for in a multidimensional manner. This includes negotiating in conjunction with the IRQ arbiter 760 and respective components that may act directly and/or indirectly with the arbiter.

Please replace the paragraph beginning on page 14, line 13 that starts with "With reference to" with the following amended paragraph:

With reference to Fig. 9, further ISA device processing is illustrated. At 910, an IRQ is selected ~~from~~ from the ISA device's possible resources. At 920 a determination is made as to whether any IRQs are left in the possible resources. If not, the process ends at 930 without satisfying the request. If so the process proceeds to 940. At 940, a determination is made as to whether an IRQ is already in use. If not, the process assigns this IRQ and a free IDT entry at 950. If yes, the process proceeds to 960. At 960, the process determines whether properties of the device that is already using the IRQ matches the properties of the new device. If not, the process proceeds back to 910 and selects another IRQ. If so the process proceeds to 970. At 970, the process determines whether both the existing device and the new device support sharing IRQs. If not, the process proceeds back to 910 and selects another IRQ. If so, the IRQ is assigned along with a free IDT entry at 980.

10/673,805MS301930.01/MSFTP381US

Please replace the paragraph beginning on page 14, line 13 that starts with "Proceeding to Fig. 11" with the following amended paragraph:

Proceeding to Fig. 11, further PCI device processing is provided for devices connected directly to an interrupt controller. At 1110 a determination is made as to whether a static IRQ is already in use. If not the process proceeds to 1120 and assigns an IRQ and IDT entry associated with a static vector. If yes at 1110, the process proceeds to 1130 and determines which processors the device's interrupts should be connected to. At 1140, a determination is made as to whether a free IDT entry is common to the respective processors. If not, the process ends at 1150 without assigning resources. If yes at 1140, the process assigns the IDT set of entries to the IRQ and the device at 1160.